DevOPS

Course Outline

Concepts	Step 1: Provisioning	Step 2: automation and Configuration Management	Step 3: Continuous Integration	Step 4: Deployment	Step 5: Continuous Monitoring	Microservices & Containerization
What is DevOps	Vagrant	Ansible	Continuous Integration	Artifactory	Nagios	The Twelve factor app
Version Control		Chef	Jenkins	Ubuntu repository	Application Monitoring	Microservices
Continuous Delivery		AWS OpsWorks	build, test, package	chef deployment	ELK stack	Containerization
Continuous Deployment						Container Orchestration
Configuration Management						Kubernetes
Provisioning & Cloud						

Mastering Ansible

ansible --list hosts (lists all hosts in the inventory file)

Ansible Tasks for troubleshooting (not playbooks; these are direct commands)
ansible -m ping all (pings all hosts; -m is for module name)
ansible -m command -a "hostname" all (finds hostnames of all hosts) or u can also use ansible -a "hostname" all since -m command is assumed.

Ansible Plays

1) Play to install nginx

```
---
 - hosts: control
   become: true                    # this is equivalent to the sudo command
   tasks:
    - name: install nginx
      yum: name=nginx state=present update_cache=yes
```

2) Installing mutiple packages

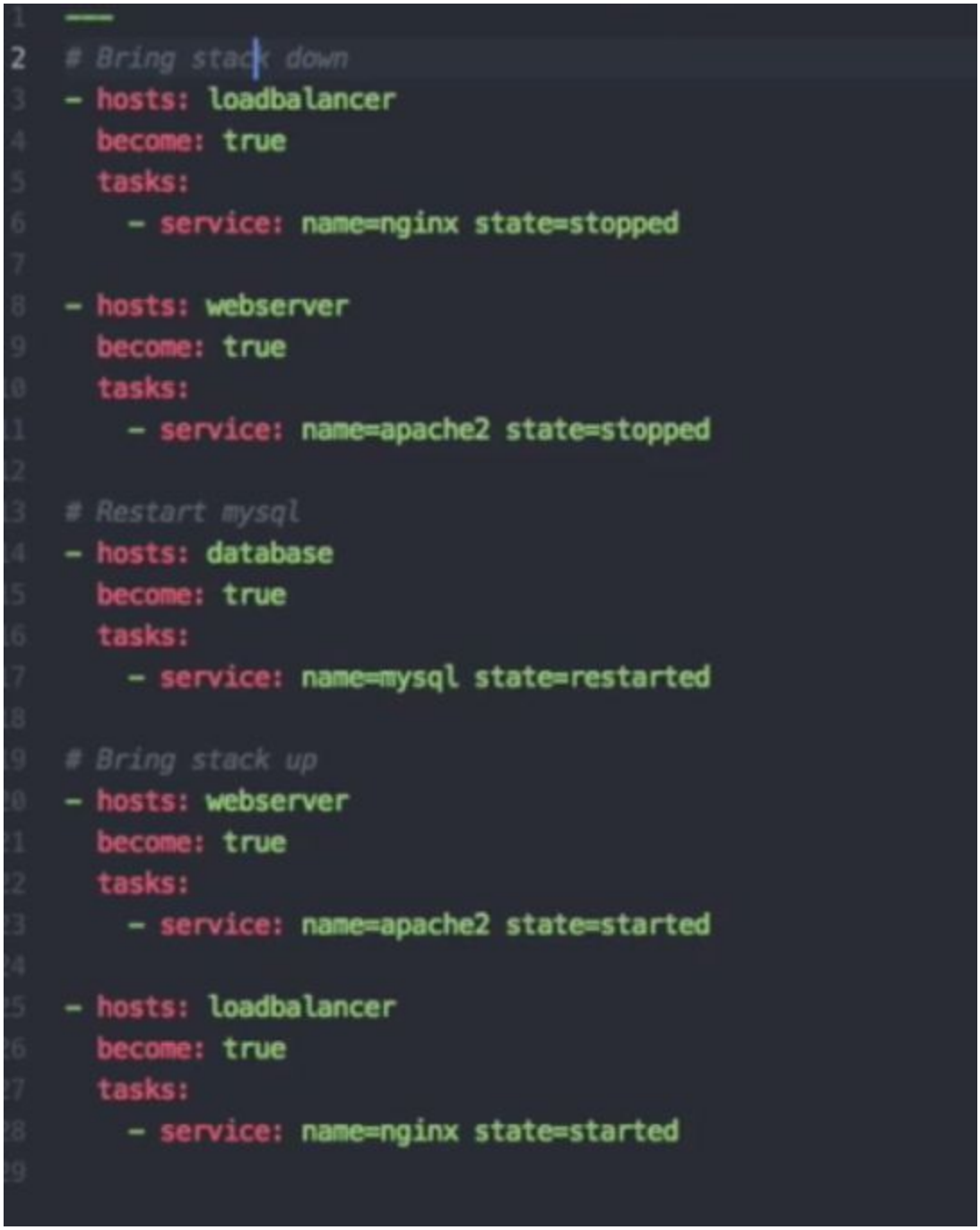
```
---
 - hosts: control
   become: true
   tasks:
    - name: install multiple packages
      yum: name={{item}} state=present update_cache=yes
      with_items:
       - apache2
       - python-pip
       - python-virtualenv
```

3) Service module

```
---
 - hosts: control
   become: true                    # this is equivalent to the sudo command
   tasks:
    - name: install nginx
      yum: name=nginx state=present update_cache=yes
    - name: ensure nginx service is started
      service: name=nginx state=started enabled=yes
```

4) to run ansible playbook
ansible-playbook test.yml

5) Stack restart example



```
---
# Bring stack down
- hosts: loadbalancer
  become: true
  tasks:
    - service: name=nginx state=stopped

- hosts: webserver
  become: true
  tasks:
    - service: name=apache2 state=stopped

# Restart mysql
- hosts: database
  become: true
  tasks:
    - service: name=mysql state=restarted

# Bring stack up
- hosts: webserver
  become: true
  tasks:
    - service: name=apache2 state=started

- hosts: loadbalancer
  become: true
  tasks:
    - service: name=nginx state=started
```

File copy

webserver.yml

```
---
- hosts: webserver
  become: true
  tasks:
    - name: install web components
      apt: name={{item}} state=present update_cache=yes
      with_items:
        - apache2
        - libapache2-mod-wsgi
        - python-pip
        - python-virtualenv

    - name: ensure apache2 started
      service: name=apache2 state=started enabled=yes

    - name: ensure mod_wsgi enabled
      apache2_module: state=present name=wsgi
      notify: restart apache2

    - name: copy demo app source
      copy: src=demo/app/ dest=/var/www/demo mode=0755
      notify: restart apache2

  handlers:
    - name: restart apache2
      service: name=apache2 state=restarted
```

Installing virtualenv using pip
Under copy demo app source

- name: setup python virtualenv
  pip: requirements=var/www/demo/requirements.txt virtualenv=var/www/demo/.venv
  notify: restart apache2

Examples (from ansible documentation)

```
- name: install the latest version of Apache
  yum: name=httpd state=latest

- name: remove the Apache package
  yum: name=httpd state=absent

- name: install the latest version of Apache from the testing repo
  yum: name=httpd enablerepo=testing state=present

- name: install one specific version of Apache
  yum: name=httpd-2.2.29-1.4.amzn1 state=present

- name: upgrade all packages
  yum: name=* state=latest

- name: install the nginx rpm from a remote repo
  yum:
name=http://nginx.org/packages/centos/6/noarch/RPMS/nginx-release-centos-6-0.el6.ngx.noarch
.rpm state=present

- name: install nginx rpm from a local file
  yum: name=/usr/local/src/nginx-release-centos-6-0.el6.ngx.noarch.rpm state=present

- name: install the 'Development tools' package group
  yum: name="@Development tools" state=present

- name: install the 'Gnome desktop' environment group
  yum: name="@^gnome-desktop-environment" state=present
```